

INTERNET DE LAS COSAS

12 de junio de 2020

DECLARACIÓN DE HONESTIDAD ACADÉMICA

Doy mi palabra ante la Universidad y la sociedad que demanda ciudadanos comprometidos con un actuar correcto, que la presente evaluación fue desarrollada con completa honestidad y responsabilidad, atendiendo a la misión institucional de formar seres íntegros y líderes que ayuden a construir una mejor sociedad

NOTA: En el quiz se permitirá tener programas desarrollados previamente y utilizarlo como plantilla, pero no se permitirán notas ni diapositivas, como tampoco el uso de internet para fines diferentes al de acceder al Moodle.

NOTA: Abajo se explica que columnas y nombres debe tener la bas de datos a crearse en MySQL, lea muy bien todo el examen antes de comenzar a solucionarlo.

INFRAESTRUCTURA DE RED (0.2 por cada pregunta) (1.0 total)

1. Se busca implementar en un hospital, una red para IoT que pueda contener hasta 65535 dispositivos por red, y permita tener hasta 65535 Vlanes diferentes, que clase de mascara de subred se necesitaría implementar?
2. Según la mascara de subred definida en el punto 1, para la primera Vlan disponible, cual sera la minima y la máxima dirección IP disponible para dispositivos?
3. Suponga que se tiene un paquete de datos, con una dirección de destinatario; Que capa del modelo OSI permitirá que el paquete llegue al destinatario correcto?
4. Si tengo un servicio de red que necesita mínima latencia y una comunicación lo mas rápida posible, que servicio de red sería adecuado?
5. Si tengo un paquete de datos que una aplicación necesita enviar, que capa del modelo OSI me permitirá seleccionar el protocolo adecuado según la aplicación?

SISTEMA DE CONTROL A DISTANCIA DE PACIENTES EN CUARENTENA

Usted ha sido contratado por una empresa medica para desarrollar un sistema de control a distancia de pacientes en cuarentena por síntomas de COVID-19.

En el examen parcial se realizó el primer desarrollo de un sistema de monitoreo a distancia, donde se tenían ciertas variables que se median para enviarlas a una central mediante comunicación serial.

Dada la escalabilidad que se espera del proyecto, se busca pasar a comunicación inalámbrica en vez de serial cableada, donde se desarrolle toda la parte de hardware nuevamente, leyendo los sensores, procesando la información y enviándola a un servidor seguro, donde los datos se guarden en una base de datos.

Al igual que en el examen parcial, se busca controlar el nivel de medicina en el paciente, mediante un actuador electromecánico que actúa como una bomba peristáltica, suministrando una cantidad de medicina según un nivel de 0 a 100%, durante un determinado tiempo T.

Adicionalmente se busca leer y controlar el nivel de oxigenación mediante un sensor de oxígeno y un compresor especial según un nivel de entre 0 y 100% de oxígeno, como también leer la temperatura de la habitación.

Por último se busca también leer si el paciente necesita atención de un auxiliar medico.

El funcionamiento esperado se describe a continuación:

- Se debe crear un Backend y Frontend, en el cual se encuentre una base de datos, donde se guarde una tabla de usuarios registrados en la pagina web; Adicionalmente se debe tener una segunda tabla donde se guarden los datos de cada paciente(El formato de las tablas se muestra en la ultima pagina).
- La pagina web debe iniciar en una pagina de login, que muestre de que se trata la pagina(Títulos adecuados, que es la aplicación, para que sirve?, adicionalmente se requiere buena presentación de la pagina), donde se encuentren dos campos de texto con usuario y contraseña para iniciar sesión con un botón de login(Se deben realizar todos los chequeos pertinentes a la hora de poder iniciar sesión correctamente, el usuario existe?, la contraseña es correcta?).

- Adicionalmente se debe tener un botón que rediriga a una pagina de registro donde se pida el nombre, correo, usuario, contraseña y confirmación de contraseña para poder registrar un usuario en la base de datos(Se debe chequear que el usuario o el correo no existan previamente en la base de datos, de lo contrario mostrar una alerta con el error correspondiente; Adicionalmente se debe chequear que las contraseñas coincidan).
- Al iniciar sesión correctamente, se debe redirigir al usuario a una pagina home, donde el usuario pueda ver los pacientes disponibles en una tabla y sus últimos valores recibidos, donde se resalte en rojo el paciente que necesite atención medica inmediata y se muestre un botón para volver al paciente al estado normal y las letras ya no se resalten en rojo(Esto quiere decir que se enviara alguien a la habitación del paciente y se resetea el estado en la base de datos).
- Adicional en la tabla de pacientes, debe haber un botón para ver el historial de cada paciente, para cuando se presione el botón, se rediriga a una pagina donde se muestren los últimos 10 datos del paciente en cuestión y la pagina muestre un titulo con el nombre del paciente en cuestión(Buscar en la tabla de pacientes todos los datos que coincidan con el nombre del paciente en cuestión y mostrar los últimos 10).
- En esta pagina debe haber un botón para regresar a la tabla de todos los pacientes.
- El sistema remoto debe enviar el estado de sus variables cada 10 segundos al servidor, donde el formato del JSON a enviar debe ser: Nombre Paciente, Nivel de medicina actual, Nivel de oxigenación, Temperatura de la habitación, necesita o no atención.
- Cuando llegue un dato de un paciente, este se debe guardar en la tabla de pacientes.
- El sistema remoto consistirá de un Arduino UNO y un ESP8266 conectado por serial.
- La bomba peristáltica es controlada mediante PWM con un nivel lógico de 0 a 5V.
- El compresor de oxígeno recibe una señal PWM con un nivel lógico de 0 a 5V.
- El sensor de nivel de oxigenación entrega una salida analoga lineal entre 0 y 5V para un nivel de 0 a 100% de nivel de oxigenación.
- El sensor de temperatura entrega una salida analoga lineal de 0 a 5V, para una temperatura de entre -55 y 105°C.

- Se debe leer toda la información de los sensores desde un Arduino UNO y enviarlos por serial al ESP con una trama adecuada, para que este lea la información y la envíe al servidor.

NOTA: La tabla de usuarios debe llamarse "usuarios" y debe tener las siguientes columnas en el siguiente orden:

- id, nombre, correo, usuario, contraseña.

La tabla de pacientes debe llamarse "pacientes" y debe tener las siguientes columnas en el siguiente orden:

- id, nombre, nivel_medicina, nivel_oxigenacion, temp_habitacion, estado_atencion.

Evaluación:

1. Crear un proyecto en Arduino IDE para el Arduino UNO que leerá las variables y las enviará por serial.**(0.75)**
2. Crear un programa en Arduino IDE para el ESP8266, el cual recibirá las variables por serial y las enviará como un vector JSON al servidor.**(0.75)**
3. Crear una instancia de NodeJS donde se almacene el código del Backend funcional con lo que se pide.**(1.25)**
4. Crear un código Frontend con todas las funciones, buena presentación y funcionalidad que se requiere.**(1.25)**
5. Responda las preguntas acerca de la teoría de infraestructura de red**(1.0)**

NOTA: Guarde los dos archivos de Arduino dentro de una carpeta, junto con una carpeta que contenga toda la instancia Backend y Frontend, adicionalmente, guarde en la carpeta un pantallazo del workbench de las tablas creadas en la base de datos.

La carpeta se debe comprimir y llamar FINALIOT_NOMBRE.zip, Envíela al correo alejandro.puerta@eia.edu.co con el asunto FINALIOT_NOMBRE.

MUCHOS ÉXITOS!!